

The Minimum Spanning Tree of Maximum Entropy

Samuel de Sousa and Walter G. Kropatsch
Pattern Recognition and Image Processing (PRIP)
Vienna University of Technology, Austria

Abstract. In computer vision, we have the problem of creating graphs out of unstructured point-sets, i.e. the data graph. A common approach for this problem consists of building a triangulation which might not always lead to the best solution. Small changes in the location of the points might generate graphs with unstable configurations and the topology of the graph could change significantly. After building the data-graph, one could apply Graph Matching techniques to register the original point-sets. In this paper, we propose a data graph technique based on the Minimum Spanning Tree of Maximum Entropy (MSTME). We aim at a data graph construction which could be more stable than the Delaunay triangulation with respect to small variations in the neighborhood of points. Our technique aims at creating data graphs which could help the point-set registration process. We propose an algorithm with a single free parameter that weighs the importance between the total weight cost and the entropy of the current spanning tree. We compare our algorithm on a number of different databases with the Delaunay triangulation.

1 Introduction

The problem of point-set registration often involves the construction of the so-called data graph [4, 7], which is the graph created out of unstructured point-sets. One possibility for registering point-sets using graphs is to formulate the problem as Graph Matching [12]. We are interested in exploiting some topological properties of data graphs which would alleviate the registration procedure. For instance, we could design a data-graph which would decrease the search space when matching points.

The Iterative Closest Point (ICP) algorithm [3] is a powerful solution for the registration of point-sets. Such an algorithm would try to find for each point in one set the closest point in the other set. Therefore, we could avoid checking all points if we have some prior knowledge. For instance, if we could state that the point we would like to match has a certain degree value, we could evaluate only the points which match this criteria and greatly reduce the search space for the registration. This would be true for a non regular data-graph ¹, otherwise we would still need to search the whole space of points.

We would like to obtain a graph with high variability in the degree distribution. This high variability can be encoded by the entropy of the degree distribution. In [5], we have performed the registration of point-sets via the Min-Weight Max-Entropy problem in which the generated edge-induced subgraph had the maximum entropy. In this work, we constrain our edge-induced subgraph to be a tree. Our problem is called the Minimum Spanning Tree with

¹A graph in which all nodes have the same degree value.

Maximum Entropy (MSTME). By minimizing the total edge weight, we can allow some robustness of the data graph with respect to small deviations of the points, by maximizing the entropy, we increase the variability of the degree distribution which could later ease the registration process. We evaluate the current approach under different levels of noise and compare the stability of the data-graph with a Delaunay triangulation.

The remainder of this paper is organized as follows: The related work is presented in Section 2. Section 3 discusses our objective function and explains how the entropy is measured in the graph. Our optimization strategy is explained in details in Section 4. Our experimental setup is disclosed in Section 5. Our conclusions and future work are available in Section 6.

2 Related Work

A data graph can be obtained in a variety of ways, it is common to use the Delaunay triangulation for the registration of point-sets [4, 7, 6]. The Delaunay triangulation is based on the condition that no other point should lie inside the circumcircle of any triangle. It is the dual of the Voronoi diagram which partitions the embedding space into regions closest to the point set. In this work, we are working on suitable alternatives which would allow more stability during the registration.

Our cost function is built upon a Minimum Spanning Tree formulation and entropy maximization. In this sense, our work is closely related to [8] as the authors also formulate an optimization problem focusing on the entropy of a Minimum Spanning Tree. However, they are dealing with a different problem by proposing an entropy estimator for clustering. In our work, we are interested on the data graph construction problem and our cost function differs from theirs since our entropy is calculated on the degree distribution of the generated MST while they estimate the data entropy based on the length of the spanning tree.

Our problem can also be seen as a bi-criteria optimization problem. Ravi and Goemans [10] proposed a Lagrangian relaxation to solve the constrained minimum spanning tree problem, which is also a bi-criteria optimization proven to be NP-Hard by Aggarwal et al. [1]. Neighborhood search and adjacency search heuristics for the bicriterion minimum spanning tree problem were proposed by Andersen et al. [2].

3 Objective Function

In order to measure the diversity found in the degree distribution, we can calculate the Shannon entropy (H) for a graph $G(V, E)$ as follows:

$$H(G) = - \sum_{v \in V_{\neq}} p(v) \log_2(p(v)). \quad (1)$$

where $p(v)$ is the probability of finding a node with a degree of v among all distinct degree values (V_{\neq} is the set of distinct degree values of V). The entropy measures the uncertainty associated with a random variable. As defined in Eq. 1, a high entropy $H(G)$ of a graph $G(V, E)$ would indicate a high “variability” in the distribution of nodes V . The converse is also true, a low entropy means low variability, as in a k -regular graph whose probability $p(k) = 1$ and $\log(1) = 0$.

Given a point-set \mathcal{X} , we calculate the distance from each point to all the other points and store those distances in W . This generates a complete graph

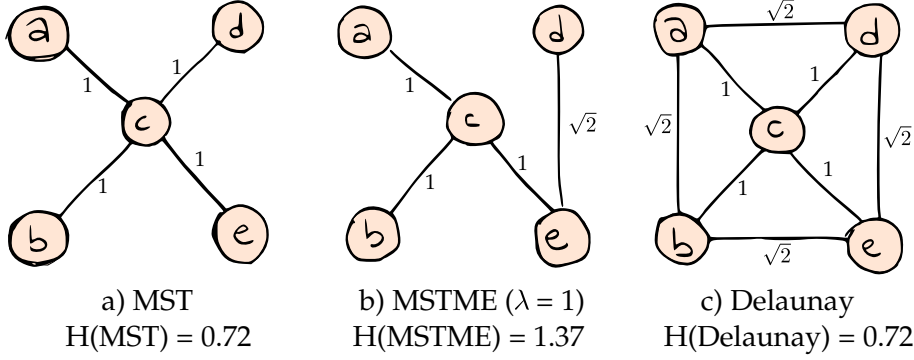


Figure 1: The first graph (a) is obtained with a Minimum Spanning Tree (MST) and has cost $c(\text{MST}) = 4$ and $H(\text{MST}) = 0.72$. Graph (b) is obtained using our approach of MST with Maximum Entropy (MSTME). Since $H(\text{MSTME}) = 1.37$, the cost $c(\text{MSTME}, \lambda = 1) = 3.04$, hence, $c(\text{MST}) < c(\text{MSTME})$. The last graph (c) is a Delaunay triangulation and it has the same entropy as the MST.

$K_{|\mathcal{X}|}(\mathcal{X}, W)$. We seek to find a vector $U \in \{0, 1\}^{|W|}$ in such a way that when $U_i = 1$, we add W_i to our graph. We can formulate our cost function simply as:

$$\begin{aligned}
 & \underset{U}{\text{minimize}} && \sum_{i=1}^{|W|} W_i U_i - \lambda(H(G[U])) \\
 & \text{subject to} && U \in \{0, 1\}^{|W|}, \\
 & && \sum U = |W| - 1, \\
 & && G[U] \text{ is connected.}
 \end{aligned} \tag{2}$$

where $G[U]$ is an edge-induced subgraph from vector U . λ is a parameter which weighs the importance between the two terms: minimizing the total edge and maximizing the graph entropy. The first constraint states that we have a discrete problem, i.e. we either add an edge ($U_i = 1$) or we do not add it. The second constraint is a necessary condition for our graph to be a tree, however, we need the third constraint to force the graph to be connected.

Figure 1 displays three different graphs of the same point-set $\mathcal{X} = \{a, b, c, d, e\}$. The first graph (Figure 1a) is simply a Minimum Spanning Tree (MST) whose cost is $c(\text{MST}, \lambda = 0) = 4$. It could be obtained by setting $\lambda = 0$ and therefore the entropy part vanishes from Equation 2. The second graph (Figure 1b) now considers $\lambda = 1$, a higher entropy would be able to decrease the cost of solution in comparison with the MST, the cost is $c(\text{MSTME}, \lambda = 1) = 3.04$. Notice that although the edge weight cost is higher, the total cost was decreased due to the higher entropy. Finally, we show in contrast that calculating a Delaunay triangulation yields an entropy as high as the MST one.

4 Optimization for the MSTME Problem

Our optimization is based on the following strategy: we pose the MSTME as a capacity problem with respect to the edge set. At each iteration, we increase the capacity of the edges, and we add the one whose cost is minimum based on our cost function which evaluates the weight and entropy of the current generated MST.

The input is a complete graph $K_{|\mathcal{X}|}(\mathcal{X}, W)$ composed of the point-set as the vertices and the edges are weighted according to the Euclidean distance. For each edge e , we check if the subgraph $G[U]$ induced by our vector U contains a cycle by the addition of such an edge, in which case, we discard (line 8). Otherwise, we calculate the new cost obtained by the weight of e and we estimate the entropy of the MST (line 12). In line 16, we remove this edge from our graph and continuing searching within the same capacity for an edge which could yield a smaller cost. At the end, we add the edge with the best cost an increase the capacity of our MST up to $|\mathcal{X}| - 1$ (line 17).

1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18	Data: $K_{ \mathcal{X} }(\mathcal{X}, W), \lambda;$ Result: $G[U];$ begin $U \leftarrow \{\};$ // We increase the capacity up to $ \mathcal{X} - 1$ for $i = 0$ to $ \mathcal{X} - 1$ do $cost \leftarrow \infty;$ $best \leftarrow 0;$ foreach $e \in E \setminus U$ do if $G[U \cup \{e\}]$ <i>has cycle</i> then continue; else $U \leftarrow U \cup \{e\};$ $ecost = W_e - \lambda H(G[U])$ if $ecost < cost$ then $cost \leftarrow ecost;$ $best \leftarrow e;$ $U \leftarrow U - \{e\};$ $U \leftarrow U \cup \{best\};$ return $G[U];$
---	---

Algorithm 1: The Minimum Spanning Tree with Maximum Entropy.

5 Experiments

Our experimental section focus on evaluating the stability of our algorithm on point-sets sampled from the silhouette of objects. Figure 2 shows some point-sets connected using the MSTME. It is noticeable that the entropy takes over in thin areas, i.e. the degree values of the points are higher in regions such as the top of the fish, the feet of the dog, while other regions follow the silhouette of the object, since it is too costly to make an edge cross the shape of the object.

Our noise experiment is built upon the shortest pairwise distance (ϵ) in the point-set. Then, we perurb each point $x \in \mathcal{X}$ by a length and an angle (l, α) , where $\alpha = [0, 360)$, and the radius $l = [0, r\epsilon]$ and r varies from $[1, 10]$, e.g. 5ϵ means that each point of the dataset was perturbed within a radius of 5 times the shortest pairwise distance in the graph. Certainly, the higher the noise, the lower the stability of the algorithms will be.

Figure 3 shows the impact of noise in the point-sets varying according to ϵ in a cow silhouette. It is not an easy task even for humans to match such points (e.g. feet). It is worth mentioning that as the algorithm searches for the optimal, it has to decide on which edges to take, therefore, a simple wrong edge

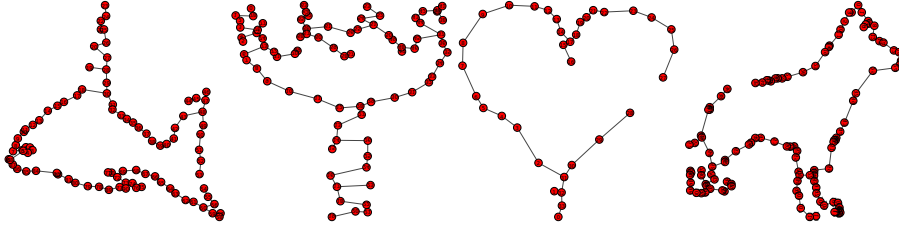
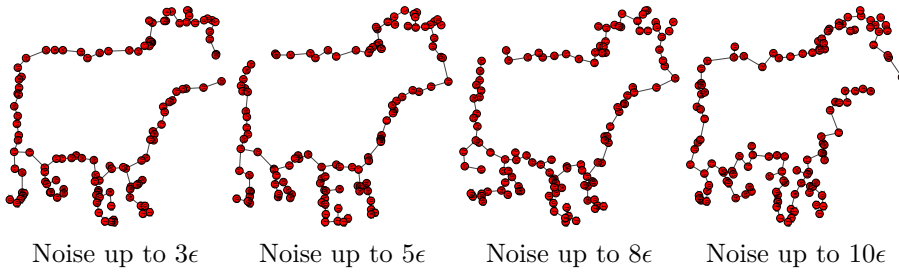
Figure 2: Data graphs connected using the MSTME ($\lambda = 0.5$).

Figure 3: Noise added to the point-set.

will cause a much higher impact on the results when compared with Delaunay triangulation, which has many more edges.

As our algorithm outputs a tree, it cannot contain cycles, therefore, the silhouette will “break” in locations where the weights of the edges are high, and the split point will vary according to the noise. Notice that the split occurred in the face of the cow when noise level was 10ϵ but in level 8ϵ it happened at a different location.

Figure 4 shows the quantitative analysis of stability between MSTME and Delaunay triangulation. The way we evaluate the stability is by calculating the percentage of edges which remained stable in all graphs within the same noise level. The first graph shows the results for the fish dataset². The y -axis displays the percentage of stable edges, e.g. 0.8 means that 80% of edges were found across all data-graphs within the same level. For this dataset, our algorithm for the MSTME was always more stable than the Delaunay triangulation up to the highest noise level. The boxplot displays several other types of information, such as the median, quartiles, etc. The instability observed was found on the thin parts of the fish, such as the dorsal and caudal fins. The second plot shows the results on the cow1 dataset³

We observed that the MSTME was, in general, more stable than the Delaunay algorithm. Small variations in the location of the nodes affected the triangulation more than it affected the MSTME. As the triangulation generates many more edges, a single wrong edge will have a high impact on the results of the MSTME. Nevertheless, it was still able to obtain higher stability than Delaunay up to a certain level of noise.

²The fish dataset was obtained from the Coherent Point Drift [9].

³the cow1 dataset was obtained from the kimia99 dataset [11].

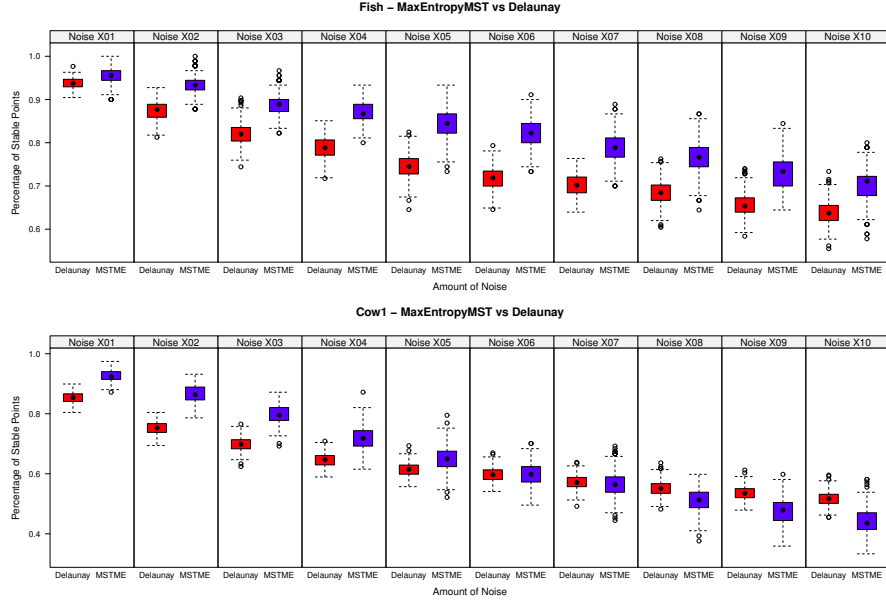


Figure 4: Boxplot comparison of the stability on the fish dataset. The noise consists of shifting each individual point by $\epsilon \times \text{level}$, varying from 1 to 10. For each noise level, we generated 30 perturbed datasets and compute the number of stable edges across all results.

6 Conclusions

In this paper, we proposed a data-graph technique by searching for a tree whose cost is minimum regarding the total edge weight and maximum regarding entropy of the degree distribution. We called this problem the Minimum Spanning Tree of Maximum Entropy (MSTME). In point-sets sampled from the silhouette of an object, we observed that points lying in thin areas were often unstable when there was noise. After analyzing and comparing the robustness of the generated graph, we notice that the data-graph generated by the MSTME was more stable than the one by the Delaunay triangulation.

In the future, we would like to propose an approximation algorithm which could guarantee bounds on the MSTME problem. Moreover, we will apply this data-graph in a framework similar to ICP and evaluate the accuracy of the results obtained as well as comparing the results using other data-graph techniques.

Acknowledgments

The first author of this paper is financially supported by the Vienna PhD School of Informatics.

References

- [1] V. Aggarwal, Y.P. Aneja, and K.P.K. Nair. Minimal spanning tree subject to a side constraint. *Computers & Operations Research*, 9(4):287 – 296, 1982. ISSN 0305-0548.

- [2] Kim Allan Andersen, Kurt Jörnsten, and Mikael Lind. On bicriterion minimal spanning trees: An approximation. *Computers & Operations Research*, 23(12):1171 – 1182, 1996. ISSN 0305-0548.
- [3] P.J. Besl and Neil D. McKay. A method for registration of 3-d shapes. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 14(2): 239–256, 1992. ISSN 0162-8828.
- [4] A. D J Cross and E.R. Hancock. Graph matching with a dual-step em algorithm. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 20(11):1236–1253, 1998. ISSN 0162-8828.
- [5] Samuel de Sousa and Walter G. Kropatsch. Data graph formulation as the minimum-weight maximum-entropy problem. In Cheng-Lin Liu, Bin Luo, Walter G. Kropatsch, and Jian Cheng, editors, *Graph-Based Representations in Pattern Recognition*, volume 9069 of *Lecture Notes in Computer Science*, pages 13–22. Springer International Publishing, 2015. ISBN 978-3-319-18223-0.
- [6] Samuel de Sousa and Walter G. Kropatsch. Graph-based point drift: Graph centrality on the registration of point-sets. *Pattern Recognition*, 48(2):368 – 379, 2015. ISSN 0031-3203.
- [7] B. Luo and E.R. Hancock. Iterative procrustes alignment with the EM algorithm. *Image and Vision Computing*, 20(5 – 6):377 – 396, 2002. ISSN 0262-8856.
- [8] Andreas C. Müller, Sebastian Nowozin, and Christoph H. Lampert. Information theoretic clustering using minimum spanning trees. In Axel Pinz, Thomas Pock, Horst Bischof, and Franz Leberl, editors, *Pattern Recognition*, volume 7476 of *Lecture Notes in Computer Science*, pages 205–215. Springer Berlin Heidelberg, 2012. ISBN 978-3-642-32716-2.
- [9] Andriy Myronenko and Xubo Song. Point Set Registration: Coherent Point Drift. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 32(12):2262–2275, December 2010. ISSN 0162-8828.
- [10] R. Ravi and M.X. Goemans. The constrained minimum spanning tree problem. In Rolf Karlsson and Andrzej Lingas, editors, *Algorithm Theory – SWAT’96*, volume 1097 of *Lecture Notes in Computer Science*, pages 66–75. Springer Berlin Heidelberg, 1996. ISBN 978-3-540-61422-7.
- [11] Daniel Sharvit, Jacky Chan, Hüseyin Tek, and Benjamin B. Kimia. Symmetry-based indexing of image databases. *Journal of Visual Communication and Image Representation*, 9(4):366–380, December 1998.
- [12] Mario Vento. A one hour trip in the world of graphs, looking at the papers of the last ten years. In Walter G. Kropatsch, NicoleM. Artner, Yll Haxhimusa, and Xiaoyi Jiang, editors, *Graph-Based Representations in Pattern Recognition*, volume 7877 of *Lecture Notes in Computer Science*, pages 1–10. Springer Berlin Heidelberg, 2013. ISBN 978-3-642-38220-8.